

**WE CLAIM:**

1. A method of exchanging information internal to a data switching node between at least two internal components, the method comprising steps of:
  - a. encapsulating the information to be exchanged in a data frame at a first component;
  - b. conveying the data frame between the first component and a second component via a data exchange medium; and
  - c. selectively decapsulating the exchanged information at the second component

whereby the conveyance of encapsulated information provides a common information exchange format facilitating expansion, upgrade and new deployment of features and services in data switching environments.

2. A method as claimed in claim 1, wherein encapsulating the exchanged information, the method further comprises a step of encapsulating a data stream.
3. A method as claimed in claim 2, wherein encapsulating the data stream, the method further comprises a step of dividing the data stream into a stream of data granules, each data frame encapsulating a single data granule.
4. A method as claimed in claim 3, wherein dividing the data stream into a stream of granules, the method further comprises a step of dividing the data stream into a stream of fixed size data granules.
5. A method as claimed in claim 3, wherein dividing the data stream into a stream of granules, the method further comprises a step of dividing the data stream into data granules of at most 32 bytes long.

6. A method as claimed in claim 3, wherein encapsulating exchanged information held in each data granule, the method further comprises a step of associating a header with each data granule, the encapsulated data granule representing a payload of the corresponding data frame.
7. A method as claimed in claim 6, wherein associating a header with each data granule, the method further comprises a step of associating a fixed size header therewith.
8. A method as claimed in claim 6, wherein associating a fixed size header with each encapsulated data granule, the method further comprises the step of associating an 8 bytes long header therewith.
9. A method as claimed in claim 6, wherein associating a header with each data granule, the method further comprises a step of associating a header having a data format.
10. A method as claimed in claim 9, wherein associating a header having a format with an encapsulated data granule, the method further comprises a step of associating a header having data fields.
11. A method as claimed in claim 10, wherein associating a header having data fields with the encapsulated data granule, the method further comprises a step of specifying a data frame type identifier unique to each data frame type, the data frame type identifier being specified in a specific data field.
12. A method as claimed in claim 10, wherein associating a header having data fields with the encapsulated data granule, the method further comprises a step of specifying a data frame sequence number, the data frame sequence number being specified in a data field.

- 0966694, 092804  
T03250, T699660
13. A method as claimed in claim 10, wherein associating a header having data fields with the encapsulated data granule, the method further comprises a step of identifying in a data field the encapsulating data frame as one of: an interrupt data frame, a request data frame and a response data frame.
  14. A method as claimed in claim 1, wherein conveying the data frame between the first component and the second component, the method further comprises a step of conveying an encapsulated request.
  15. A method as claimed in claim 1, wherein conveying the data frame between the first component and the second component, the method further comprises a step of conveying an encapsulated response.
  16. A method as claimed in claim 13, wherein the subsequent to conveying the encapsulated request, the method further comprises a step of conveying at least one encapsulated response generated by the request.
  17. A method as claimed in claim 16, wherein the method further comprises a step of assigning a data frame sequence number to the data frame encapsulating the response, the data frame sequence number having the same value as a data frame sequence number of the data frame encapsulating the corresponding request.
  18. A method as claimed in claim 1, wherein conveying the data frame between the first component and the second component, the method further comprises a step of conveying an encapsulated interrupt request.
  19. A method as claimed in claim 18, wherein the subsequent to conveying the encapsulated interrupt request, the method further comprises a step of conveying at least one encapsulated request generated by the interrupt request.

- 09966691-092801  
T092801-09966691
20. A method as claimed in claim 19, wherein the method further comprises a step of assigning a data frame sequence number to the data frame encapsulating the request, the data frame sequence number having the same value as a data frame sequence number of the data frame encapsulating the interrupt request.
  21. A method as claimed in claim 18, wherein the subsequent to conveying the encapsulated interrupt request, the method further comprises a step of conveying at least one encapsulated response generated by the interrupt request.
  22. A method as claimed in claim 21, wherein the method further comprises a step of assigning a data frame sequence number to the data frame encapsulating the response, the data frame sequence number having the same value as a data frame sequence number of the data frame encapsulating the interrupt request.
  23. A method as claimed in claim 10, wherein associating a header having data fields with the encapsulated data granule, the method further comprises a step of identifying in a data field data field memory addressing used in conveying data to a specified memory address.
  24. A method as claimed in claim 10, wherein associating a header having data fields with the encapsulated data granule, the method further comprises a step of identifying in a data field data field memory addressing used in conveying data from a specified memory address.
  25. A method as claimed in claim 1, wherein encapsulating the exchanged information, the method further comprises a step of encapsulating the exchanged information in a payload portion of the data frame, the payload portion of the data frame having a data format.

26. A method as claimed in claim 1, wherein encapsulating the exchanged information, the method further comprises a step of encapsulating the exchanged information in at least one data field of a payload portion of the data frame, the at least one data field having a fixed position within the data frame.
27. A method as claimed in claim 26, wherein conveying the data frame, the method further comprises a step of conveying empty fields as data padding.
28. A method as claimed in claim 1, wherein conveying the data frame, the method further comprises a step of signaling a beginning of the conveyance of the data frame.
29. A method as claimed in claim 1, wherein conveying the data frame, the method further comprises a step of signaling an end of the conveyance of the data frame.
30. A method as claimed in claim 29, wherein conveying the data frame, the method further comprises a step of signaling the end of the data frame as soon as all relevant exchanged information has been conveyed whereby variable length data frames are conveyed enabling an efficient use of the bandwidth of the data exchange medium.
31. A method as claimed in claim 1, wherein conveying the data frame, the method further comprising a step of dividing the data frame into data fragments for transmission over the data exchange medium.
32. A method as claimed in claim 31, wherein dividing the data frame into data fragments, the method further comprises a step of dividing the data frame into data fragments having a length of 8 bits for transmission via a data exchange medium comprising an 8 bit data bus.

33. A method as claimed in claim 31, wherein dividing the data frame into data fragments, the method further comprises a step of dividing the data frame into data fragments having a length of 16 bits for transmission via a data exchange medium comprising a 16 bit data bus.
34. A method as claimed in claim 31, wherein dividing the data frame into data fragments, the method further comprises a step of dividing the data frame into bits for transmission via a data exchange medium comprising an serial link.